Unconstrained Scalable Test Problems for Single-Objective Bilevel Optimization

Ankur Sinha, Pekka Malo and Kalyanmoy Deb
Department of Information and Service Economy
Aalto University School of Economics
PO Box 21220, 00076 Aalto, Helsinki, Finland
{Firstname.Lastname}@aalto.fi

KanGAL Report Number 2012003

***Abstract*— In this paper, we propose a set of six test problems for single-objective bilevel optimization. The test-collection represents various difficulties which are commonly encountered in practical bilevel optimization problems. To support experiments with problems of different size, all of the test problems are scalable in terms of the number of variables. The problem set is also accompanied by a construction procedure, which helps to generate new test problems with controlled difficulties in convergence and interaction patterns between the two optimization levels. To provide a baseline result for easy comparisons, we have solved a 10 variable instance for each of the test problems using a simple bilevel evolutionary algorithm. The results presented may be used as a benchmark while evaluating the performance of any bilevel optimization algorithm.**

***Index Terms*— Bilevel optimization, test problem construction, bilevel test-suite, bilevel evolutionary algorithm.**

I. Introduction

Bilevel optimization is a branch, which deals with optimization problems containing an additional optimization problem within the constraints. Such problems arise in many practical contexts, such as transportation (network design, optimal pricing), economics (Stackelberg games, principal-agent problem, taxation, policy decisions), management (network facility location, coordination of multi-divisional firms), engineering (optimal design, optimal chemical equilibria) etc [5]. Inspite of a large number of applications, the real life implementations are scarce [3] due to the lack of efficient algorithms which could handle generic bilevel problems. Moreover, there does not exist a test-bed on which some of the existing procedures could be evaluated systematically.

The aim of this paper is to propose a systematic framework for constructing bilevel test problems with controlled difficulties. Such test problems are necessary to evaluate the performance of any algorithm, or make a comparison between the performance of different algorithms. The test problems should be able to represent the difficulties which practical application problems might have in store for the algorithms. Moreover, the difficulties should be controllable, in order to assess the performance on different difficulty frontiers. Past studies [6] on bilevel optimization have introduced test problems where the difficulty level of the problems cannot be controlled. In most of the studies, the problems are linear [7], or quadratic [1], [2], or non-scalable with fixed number of decision variables, or too complex such that the true optimal solution is not known. These drawbacks pose difficulties in algorithm development, as the performance of the algorithms cannot be evaluated on different difficulty frontiers. The contribution of this paper is to propose a collection of scalable bilevel test problems along with a simple test problem construction framework. The construction procedure allows to control the difficulties at the two levels independently, and also allows to control the difficulty caused by the interaction of the two levels. The problems generated by the framework are such that the optimal solution of the overall bilevel problem along with the optimal solution(s) for lower level problem are known for any set of upper level variables. This makes interpretation of the results easier and helps the algorithm developers to debug their procedure during the development phase.

The paper is organized as follows. In the next section, we explain the structure of a general bilevel optimization problem and introduce central notation that is used throughout the paper. Section III presents our framework for constructing scalable test problems for bilevel programming. Thereafter, following the guidelines of the construction procedure, we suggest a set of six scalable test problems. A summary of the problems is given in Section IV. To create a benchmark for evaluating different solution algorithms, the problems are solved using a simple bilevel evolutionary algorithm which is a nested scheme described in Section V. The results for the baseline algorithm are discussed in Section VI.

II. Description of a Bilevel Problem

A bilevel optimization problem is a hierarchical optimization problem which has two levels of optimization tasks. The structure of a bilevel optimization demands that only the optimal solutions of the lower level optimization problem are acceptable as feasible members for the upper level optimization problem. The problem contains two classes of variables, the upper level variables $\mathbf{x}_u$, and the lower level variables $\mathbf{x}_l$. For the lower level problem, the optimization task is performed with respect to variables $\mathbf{x}_l$, and the variables $\mathbf{x}_u$ act as a parameter. A different $\mathbf{x}_u$ leads to a different lower level optimization problem whose optimal solution needs to be determined. The upper level problem usually involves all variables $\mathbf{x} = (\mathbf{x}_u, \mathbf{x}_l)$, and the optimization is expected to be performed with respect to both the sets of variables. A general bilevel optimization problem with one objective at both levels can be described as follows:

Definition 1 (Bilevel Optimization Problem (BLOP)): Let $X = X_U \times X_L$ denote the product of the upper-level decision space X_U and the lower-level decision space X_L, i.e. $\mathbf{x} = (\mathbf{x}_u, \mathbf{x}_l) \in X$, if $\mathbf{x}_u \in X_U$ and $\mathbf{x}_l \in X_L$. For upper-level objective function $F : X \to \mathbb{R}$ and lower-level objective function $f : X \to \mathbb{R}$, a general bilevel optimization problem is given by

$$\begin{aligned} \underset{\mathbf{x} \in X}{\text{Minimize}} \quad & F(\mathbf{x}), \\ \text{s.t.} \quad & \mathbf{x}_l \in \underset{\mathbf{x}_l \in X_L}{\operatorname{argmin}} \left\{ f(\mathbf{x}) \;\middle|\; g_i(\mathbf{x}) \geq 0, i \in I \right\}, \\ & G_j(\mathbf{x}) \geq 0, j \in J. \end{aligned} \tag{1}$$

where the functions $g_i : X \to \mathbb{R}$, $i \in I$, represent lower-level constraints and $G_j : X \to \mathbb{R}$, $j \in J$, is the collection of upper-level constraints.

A variable vector $\mathbf{x}^\star = (\mathbf{x}_u^\star, \mathbf{x}_l^\star)$ is a feasible member for the upper level only if it satisfies all the upper level constraints, and vector $\mathbf{x}_l^\star$ is an optimal solution to the lower level problem with parameters $\mathbf{x}_u^\star$. Therefore, to emphasize the nature of the lower-level problem as a parametrized constraint to the upper-level problem, an equivalent formulation of the bilevel optimization problem is obtained by replacing the lower-level minimization problem with a set value function which maps the given upper-level decision vector to the corresponding set of optimal lower-level solutions. In the language inspired by Stackelberg's games, such mapping is often called the rational reaction of the follower on the leader's choice $\mathbf{x}_u$.

Definition 2 (Alternative definition of Bilevel Problem): Let set-valued function $\Psi : X_U \rightrightarrows X_L$, denote the optimal-solution set mapping of the lower level problem, i.e.

$$\Psi(\mathbf{x}_u) = \underset{\mathbf{x}_l \in X_L}{\operatorname{argmin}} \left\{ f(\mathbf{x}) \;\middle|\; g_i(\mathbf{x}) \geq 0, i \in I \right\}.$$

A general bilevel optimization problem (BLOP) is then given by

$$\begin{aligned} \underset{\mathbf{x} \in X}{\text{Minimize}} \quad & F(\mathbf{x}), \\ \text{s.t.} \quad & \mathbf{x}_l \in \Psi(\mathbf{x}_u), \\ & G_j(\mathbf{x}) \geq 0, j \in J. \end{aligned} \tag{2}$$

where the function Ψ may be a single-vector valued or a multi-vector valued function depending on whether the lower level function has multiple global optimal solutions or not.

In this paper, we utilize the Ψ function in the test problem construction procedures to provide a convenient description of the relationship between the upper and lower level problems. To illustrate the behavior of Ψ mapping, Figures 1 and 2 show the two scenario where Ψ can be a single vector valued or a multi-vector valued function respectively. In Figure 1, the lower level problem is a paraboloid with a single minimum function value corresponding to the set of upper level variables $\mathbf{x}_u$. On the other hand, in Figure 2, the lower level function is a paraboloid sliced from the bottom with a horizontal plane. This leads to multiple minimum values for the lower level problem, and therefore, multiple lower level solutions correspond to each set of upper level variables $\mathbf{x}_u$.

III. Test Problem Construction Procedure

The presence of an optimization problem within the constraints leads to a multi-fold increase in complexity for solving bilevel optimization tasks as compared to common optimization problems. In order to create realistic test problems for bilevel optimization, the construction procedure should be able to induce difficulties at both levels independently and collectively, such that the performance of algorithms in handling the two levels is evaluated. Moreover, the problems are expected to be scalable in terms of number of decision variables to evaluate the performance of the algorithms against increasing number of variables. Some of the desired properties in the test problems along with the ones already mentioned are:

1) Controlled difficulty in convergence at upper and lower levels.
2) Controlled difficulty caused by interaction of the two levels.
3) Multiple global solutions at the lower level for any given set of upper level variables.
4) Clear identification of a relationship between the lower level optimal solutions and the upper level variables.
5) Scalability to any number of decision variables at upper and lower levels.
6) Possibility to have conflict or cooperation at the two levels.
7) The optimal solution of the bilevel optimization should be known.

In this paper, we propose a bilevel test-problem construction procedure, which should be able to incorporate all the above mentioned features. The current procedure is designed for an unconstrained set of test problems, and therefore we have omitted the discussion on constraints in our procedure. The construction procedure can also be extended to constrained problems, however, we leave it for a future work.

A. Overview of the test-problem framework

To create a tractable framework for test-problem construction, we split the upper and lower level functions into three components. Each of the components is specialized for induction of certain kinds of difficulties into the bilevel problem. The functions are determined by the required complexities at upper and lower levels independently, and also by the required complexities because of the interaction of the two levels. We write a generic bilevel test problem as follows:

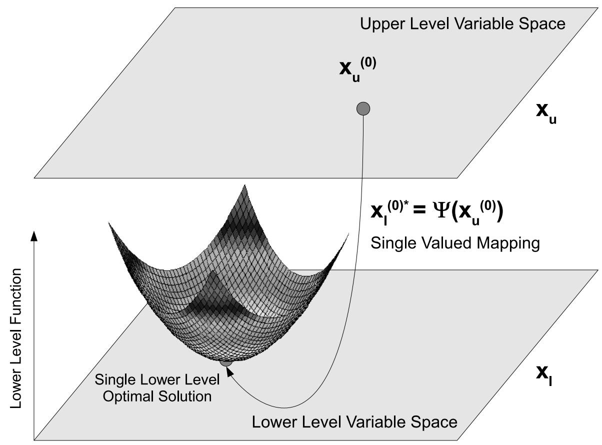

Fig. 1. Relationship between upper and lower level variables in case of a single-vector valued mapping. The lower level function is in the shape of a paraboloid.

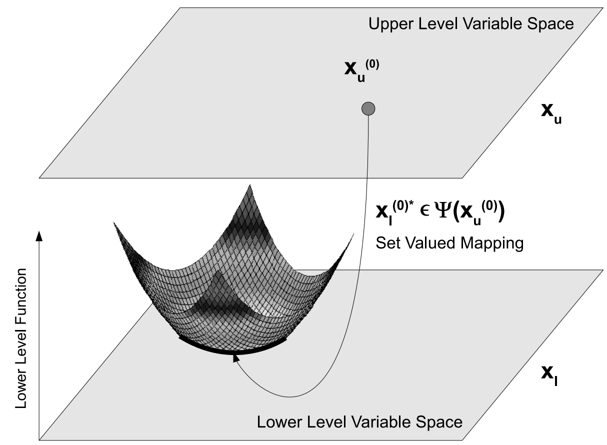

Fig. 2. Relationship between upper and lower level variables in case of a multi-vector valued mapping. The lower level function is in the shape of a paraboloid with the bottom sliced with a plane.

$$
\begin{array}{l}
F(\mathbf{x}_u, \mathbf{x}_l) = F_1(\mathbf{x}_{u1}) + F_2(\mathbf{x}_{l1}) + F_3(\mathbf{x}_{u2}, \mathbf{x}_{l2}) \\
f(\mathbf{x}_u, \mathbf{x}_l) = f_1(\mathbf{x}_{u1}, \mathbf{x}_{u2}) + f_2(\mathbf{x}_{l1}) + f_3(\mathbf{x}_{u2}, \mathbf{x}_{l2}) \\
\text{where} \\
\qquad \mathbf{x}_u = (\mathbf{x}_{u1}, \mathbf{x}_{u2}) \quad \text{and} \quad \mathbf{x}_l = (\mathbf{x}_{l1}, \mathbf{x}_{l2})
\end{array} \tag{3}
$$

In the above equations, each of the levels contains three terms. A summary on the roles of different terms is provided in Table I. The upper level and lower level variables can be seen to be broken into two smaller vectors (see Panel A in Table I). The vectors $\mathbf{x}_{u1}$ and $\mathbf{x}_{l1}$ are used to induce complexities at the upper and lower levels independently. The vectors $\mathbf{x}_{u2}$ and $\mathbf{x}_{l2}$ are responsible to induce complexities because of interaction. In similar fashion, we decompose the upper and lower level functions such that each of the components is specialized for a certain purpose only (see Panel B in Table I). At the upper level, the term $F_1(\mathbf{x}_{u1})$ is responsible for inducing difficulty in convergence solely at the upper level. Similarly, at the lower level, the term $f_2(\mathbf{x}_{l1})$ is responsible for inducing difficulty in convergence solely at the lower level. The term $F_2(\mathbf{x}_{l1})$ decides if there is a conflict or a cooperation between the upper and lower levels. The terms $F_3(\mathbf{x}_{l2}, \mathbf{x}_{u2})$ and $f_3(\mathbf{x}_{l2}, \mathbf{x}_{u2})$ are interaction terms which can be used to induce difficulties because of interaction at the two levels. Term $F_3(\mathbf{x}_{l2}, \mathbf{x}_{u2})$ may also induce a cooperation or a conflict. Finally, $f_1(\mathbf{x}_{u1}, \mathbf{x}_{u1})$ is a fixed term for the lower level optimization problem and does not induce any convergence difficulties. It is used along with the lower level interaction term to create a functional dependence between lower level optimal solution(s) and the upper level variables.

B. *Controlled difficulty in convergence*

The test-problem framework allows a number of ways to induce difficulties in the convergence of the optimization problem while retaining sufficient control. To demonstrate this, let us consider the structure of the lower level minimization problem.

For a given $\mathbf{x}_u = (\mathbf{x}_{u1}, \mathbf{x}_{u2})$, the lower level minimization problem is written as

$$
\underset{(\mathbf{x}_{l1}, \mathbf{x}_{l2})}{\text{Min}} f(\mathbf{x}_u, \mathbf{x}_l) = f_1(\mathbf{x}_{u1}, \mathbf{x}_{u2}) + f_2(\mathbf{x}_{l1}) + f_3(\mathbf{x}_{u2}, \mathbf{x}_{l2}),
$$

where the upper level variables $(\mathbf{x}_{u1}, \mathbf{x}_{u2})$ act as parameters for the optimization problem. The corresponding optimal-set mapping is given by

$$
\Psi(\mathbf{x}_u) = \text{argmin}\{f_2(\mathbf{x}_{l1}) + f_3(\mathbf{x}_{u2}, \mathbf{x}_{l2}) : \mathbf{x}_l \in X_L\},
$$

where f_1 does not appear due to its independence from $\mathbf{x}_l$. Since all of the terms are independent of each other, we note that the optimal value of the function f can be recovered by optimizing the functions f_2 and f_3 individually. Hence, as the following example shows, calibration of the desired difficulty level for the lower level problem boils down to the choice of functions f_2 and f_3 such that their optima are known.

Example 1: To create a simple lower level function, let the dimension of the variable sets be as follows: $dim(\mathbf{x}_{u1}) = U1$, $dim(\mathbf{x}_{u2}) = U2$, $dim(\mathbf{x}_{l1}) = L1$ and $dim(\mathbf{x}_{l2}) = L2$. Consider a special case where $L2 = U2$, then the three functions could be defined as follows:

$$
\begin{array}{l}
f_1(\mathbf{x}_{u1}, \mathbf{x}_{u2}) = \sum_{i=1}^{U1}(x_{u1}^i)^2 + \sum_{i=1}^{U2}(x_{u2}^i)^2 \\
f_2(\mathbf{x}_{l1}) = \sum_{i=1}^{L1}(x_{l1}^i)^2 \\
f_3(\mathbf{x}_{u2}, \mathbf{x}_{l2}) = \sum_{i=1}^{U2}(x_{u2}^i - x_{l2}^i)^2
\end{array}
$$

where f_1 affects only the value of the function without inducing any convergence difficulties. The corresponding optimal set mapping Ψ is reduced to an ordinary vector valued function

$$
\Psi(\mathbf{x}_u) = \{(\mathbf{x}_{l1}, \mathbf{x}_{l2}) : \mathbf{x}_{l1} = \mathbf{0}, \mathbf{x}_{l2} = \mathbf{x}_{u2}\}.
$$

TABLE I

OVERVIEW OF TEST-PROBLEM FRAMEWORK COMPONENTS

Panel A: Decomposition of decision variables

Upper-level variables		Lower-level variables	
Vector	Purpose	Vector	Purpose
$\mathbf{x}_{u1}$	Complexity on upper-level	$\mathbf{x}_{l1}$	Complexity on lower-level
$\mathbf{x}_{u2}$	Interaction with lower-level	$\mathbf{x}_{l2}$	Interaction with upper-level

Panel B: Decomposition of objective functions

Upper-level objective function		Lower-level objective function	
Component	Purpose	Component	Purpose
$F_1(\mathbf{x}_{u1})$	Difficulty in convergence	$f_1(\mathbf{x}_{u1}, \mathbf{x}_{u2})$	Functional dependence
$F_2(\mathbf{x}_{l1})$	Conflict / co-operation	$f_2(\mathbf{x}_{l1})$	Difficulty in convergence
$F_3(\mathbf{x}_{u2}, \mathbf{x}_{l2})$	Difficulty in interaction	$f_3(\mathbf{x}_{u2}, \mathbf{x}_{l2})$	Difficulty in interaction

As discussed above, other functions can be chosen with desired complexities to induce difficulties at the lower level and come up with a variety of lower level functions.

C. Controlled difficulty in interaction

Next, we consider the formulation of the upper level function such that a desired difficulty level in interaction between upper and lower level problems can be achieved. After having designed the lower level problem, the upper level optimization task is defined as a minimization problem over the graph of the optimal solution set mapping Ψ, i.e.

$$\text{Min } \{F(\mathbf{x}_u, \mathbf{x}_l) : \mathbf{x}_l \in \Psi(\mathbf{x}_u), \mathbf{x}_u \in X_U\}$$

where the objective function

$$F(\mathbf{x}_u, \mathbf{x}_l) = F_1(\mathbf{x}_{u1}) + F_2(\mathbf{x}_{l1}) + F_3(\mathbf{x}_{u2}, \mathbf{x}_{l2})$$

is a sum of three independent terms. Once again, our primary interest is on the last two terms $F_2(\mathbf{x}_{l1})$ and $F_3(\mathbf{x}_{u2}, \mathbf{x}_{l2})$, which determine the type of interaction there is going to be between the optimization problems. This can be done in two different ways, depending on whether a cooperation or a conflict is desired between the upper and lower level problems.

Definition 3 (Co-operative test-problem): A bilevel optimization problem is said to be co-operative, if an improvement in the lower level function leads to an improvement in the upper level function. Within our test problem framework, the independence of terms in the upper level objective function F implies that the co-operative condition is satisfied when for any upper level decision $\mathbf{x}_u$ the corresponding lower level decision $\mathbf{x}_l = (\mathbf{x}_{l1}, \mathbf{x}_{l2})$ is such that $\mathbf{x}_{l1} \in \text{argmin}\{F_2(\mathbf{x}_{l1}) : \mathbf{x}_l \in \Psi(\mathbf{x}_u)\}$ and $\mathbf{x}_{l2} \in \text{argmin}\{F_3(\mathbf{x}_{u2}, \mathbf{x}_{l2}) : \mathbf{x}_l \in \Psi(\mathbf{x}_u)\}$.

Definition 4 (Conflicting test-problem): A bilevel optimization problem is said to be conflicting, if an improvement in the lower-level function leads to an adverse effect on the upper level function's value. In our framework, a conflicting test problem is obtained when for any upper level decision $\mathbf{x}_u$ the corresponding lower level decision $\mathbf{x}_l = (\mathbf{x}_{l1}, \mathbf{x}_{l2})$ is such that $\mathbf{x}_{l1} \in \text{argmax}\{F_2(\mathbf{x}_{l1}) : \mathbf{x}_l \in \Psi(\mathbf{x}_u)\}$ and $\mathbf{x}_{l2} \in \text{argmax}\{F_3(\mathbf{x}_{u2}, \mathbf{x}_{l2}) : \mathbf{x}_l \in \Psi(\mathbf{x}_u)\}$.

In the above general form, the functions f_2 and f_3 may have multiple optimal solutions for any given upper level decision $\mathbf{x}_u$. However, in order to create test problems with tractable interaction patterns, we would like to define them such that each problem has only a single lower lower level optimum for a given $\mathbf{x}_u$. To ensure the existence of single lower level optimum, and to enable realistic interactions between the two levels, we consider imposing the following simple restrictions on the objective functions:

Case 1. Creating co-operative interaction: A test problem with co-operative interaction pattern can be created by choosing

$$\begin{aligned} F_2(\mathbf{x}_{l1}) &= f_2(\mathbf{x}_{l1}) \\ F_3(\mathbf{x}_{u2}, \mathbf{x}_{l2}) &= F_4(\mathbf{x}_{u2}) + f_3(\mathbf{x}_{u2}, \mathbf{x}_{l2}), \end{aligned} \tag{4}$$

where $F_4(\mathbf{x}_{u2})$ is any function of $\mathbf{x}_{u2}$ whose minimum is known.

Case 2. Creating conflicting interaction: A test problem with a conflict between the two levels can be created by simply changing the signs of terms f_2 and f_3 on the right hand side in (4):

$$\begin{aligned} F_2(\mathbf{x}_{l1}) &= -f_2(\mathbf{x}_{l1}) \\ F_3(\mathbf{x}_{u2}, \mathbf{x}_{l2}) &= F_4(\mathbf{x}_{u2}) - f_3(\mathbf{x}_{u2}, \mathbf{x}_{l2}). \end{aligned} \tag{5}$$

The choice of F_2 and F_3 suggested here is a special case, and there can be many other ways to achieve conflict or co-operation using the two functions.

Case 3. Creating mixed interaction: There may be a situation of both cooperation and conflict if functions F_2 and F_3 are chosen with opposite signs as,

$$\begin{aligned} F_2(\mathbf{x}_{l1}) &= f_2(\mathbf{x}_{l1}) \\ F_3(\mathbf{x}_{u2}, \mathbf{x}_{l2}) &= F_4(\mathbf{x}_{u2}) - f_3(\mathbf{x}_{u2}, \mathbf{x}_{l2}) \end{aligned} \tag{6}$$

or

$$\begin{aligned} F_2(\mathbf{x}_{l1}) &= -f_2(\mathbf{x}_{l1}) \\ F_3(\mathbf{x}_{u2}, \mathbf{x}_{l2}) &= F_4(\mathbf{x}_{u2}) + f_3(\mathbf{x}_{u2}, \mathbf{x}_{l2}). \end{aligned} \tag{7}$$

Example 2: Consider a bilevel optimization problem where the lower level task is given by Example 1. According to the above procedures, we can produce a test problem with a conflict between the upper and lower level by defining the upper level objective function as follows:

$$\begin{array}{l} F_1(\mathbf{x}_{u1}) = \sum_{i=1}^{U1}(x_{u1}^i)^2 \\ F_2(\mathbf{x}_{l1}) = -\sum_{i=1}^{L1}(x_{l1}^i)^2 \\ F_3(\mathbf{x}_{u2},\mathbf{x}_{l2}) = -\sum_{i=1}^{U2}(x_{u2}^i - x_{l2}^i)^2. \end{array} \tag{8}$$

The chosen formulation corresponds to Case 2, where $F_4(\mathbf{x}_{u2}) = 0$. The final optimal solution of the bilevel problem is $F(\mathbf{x}_u, \mathbf{x}_l) = 0$ for $(\mathbf{x}_u, \mathbf{x}_l) = \mathbf{0}$.

D. Controlled multimodality

In this sub-section, we discuss about constructing test problems with lower level function having multiple global solutions for a given set of upper level variables. To achieve this, we formulate a lower level function which has multiple lower level optima for a given $\mathbf{x}_u$, such that $\mathbf{x}_l^* \in \Psi(\mathbf{x}_u)$. Then, we ensure that out of all these possible lower level optimal solutions, one of them ($\mathbf{x}_l^{**}$) corresponds to the best upper level function value, i.e.,

$$\mathbf{x}_l^{**} \in \underset{\mathbf{x}_l^*}{\operatorname{argmin}} \left\{ F(\mathbf{x}_u, \mathbf{x}_l^*) \;\middle|\; \mathbf{x}_l^* \in \Psi(\mathbf{x}_u) \right\} \tag{9}$$

To incorporate this difficulty in the problem, we have chosen only the second functions at the upper and lower levels. The design of other functions are chosen to be done in the same way as suggested before. Given that the term $f_2(\mathbf{x}_{l1})$ is responsible for causing complexities only at the lower level, we can freely formulate it such that it has multiple lower level optimal solutions. From this it necessarily follows that the entire lower level function has multiple optimal solutions.

Example 3: We describe the construction procedure by considering a simple example, where the cardinalities of the variables are, $dim(\mathbf{x}_{u1}) = 2$, $dim(\mathbf{x}_{u2}) = 2$, $dim(\mathbf{x}_{l1}) = 2$ and $dim(\mathbf{x}_{l2}) = 2$, and the lower level function is defined as follows,

$$\begin{array}{l} f_1(\mathbf{x}_{u1},\mathbf{x}_{u2}) = (x_{u1}^1)^2 + (x_{u1}^2)^2 + (x_{u2}^1)^2 + (x_{u2}^2)^2 \\ f_2(\mathbf{x}_{l1}) = (x_{l1}^1 - x_{l1}^2)^2 \\ f_3(\mathbf{x}_{u2},\mathbf{x}_{l2}) = (x_{u2}^1 - x_{l2}^1)^2 + (x_{u2}^2 - x_{l2}^2)^2 \end{array} \tag{10}$$

Here, we observe that $f_2(\mathbf{x}_{l1})$ induces multiple optimal solutions, as its minimum value is 0 for all $x_{l1}^1 = x_{l1}^2$. At the minimum $f_3(\mathbf{x}_{u2}, \mathbf{x}_{l2})$ fixes the values of x_{l2}^1 and x_{l2}^2 to x_{u2}^1 and x_{u2}^2 respectively. Next, we write the upper level function ensuring that out of the set $x_{l1}^1 = x_{l1}^2$, one of the solutions is best at upper level.

$$\begin{array}{l} F_1(\mathbf{x}_{u1}) = (x_{u1}^1)^2 + (x_{u1}^2)^2 \\ F_2(\mathbf{x}_{l1}) = (x_{l1}^1)^2 + (x_{l1}^2)^2 \\ F_3(\mathbf{x}_{u2},\mathbf{x}_{l2}) = (x_{u2}^1 - x_{l2}^2)^2 + (x_{u2}^2 - x_{l2}^2)^2 \end{array} \tag{11}$$

The formulation of $F_2(\mathbf{x}_{l1})$, as sum of squared terms ensures that $x_{l1}^1 = x_{l1}^2 = 0$ provides the best solution at the upper level for any given $(\mathbf{x}_{u1}, \mathbf{x}_{u2})$.

IV. SMD TEST PROBLEMS

By adhering to the design principles introduced in the previous section, we now propose a set of six test problems which we call as the SMD test problems. Each problem represents a different difficulty level in terms of convergence, complexity of interaction and lower level multimodality.

A. SMD1

This is a simple test problem with cooperation between the two levels. The lower level optimization problem is a convex optimization task. The upper level is convex with respect to upper level variables and optimal lower level variables.

$$\begin{array}{l} F1 = \sum_{i=1}^{p}(x_{u1}^i)^2 \\ F2 = \sum_{i=1}^{q}(x_{l1}^i)^2 \\ F3 = \sum_{i=1}^{r}(x_{u2}^i)^2 + \sum_{i=1}^{r}(x_{u2}^i - \tan x_{l2}^i)^2 \\ \\ f1 = \sum_{i=1}^{p}(x_{u1}^i)^2 \\ f2 = \sum_{i=1}^{q}(x_{l1}^i)^2 \\ f3 = \sum_{i=1}^{r}(x_{u2}^i - \tan x_{l2}^i)^2 \end{array} \tag{12}$$

The range of variables is as follows,

$$\begin{array}{l} x_{u1}^i \in [-5,10], \;\; \forall \;\; i \in \{1,2,\ldots,p\} \\ x_{u2}^i \in [-5,10], \;\; \forall \;\; i \in \{1,2,\ldots,r\} \\ x_{l1}^i \in [-5,10], \;\; \forall \;\; i \in \{1,2,\ldots,q\} \\ x_{l2}^i \in (\frac{-\pi}{2}, \frac{\pi}{2}), \;\; \forall \;\; i \in \{1,2,\ldots,r\} \end{array} \tag{13}$$

Relationship between upper level variables and lower level optimal variables is given as follows,

$$\begin{array}{l} x_{l1}^i = 0, \;\; \forall \;\; i \in \{1,2,\ldots,p\} \\ x_{l2}^i = \tan^{-1} x_{u2}^i, \;\; \forall \;\; i \in \{1,2,\ldots,r\} \end{array} \tag{14}$$

The values of the variables at the optima are $\mathbf{x}_u = 0$ and $\mathbf{x}_l$ is obtained by the relationship given above. Both the upper and lower level functions are equal to 0 at the optima.

B. SMD2

In this test problem there is a conflict between the upper level and lower level optimization task. The lower level optimization problem is a convex optimization task. An inaccurate lower level optimum may lead to upper level function value better than the true optimum for the bilevel problem. The upper level is convex with respect to upper level variables and optimal lower level variables.

$$\begin{array}{l} F1 = \sum_{i=1}^{p}(x_{u1}^i)^2 \\ F2 = -\sum_{i=1}^{q}(x_{l1}^i)^2 \\ F3 = \sum_{i=1}^{r}(x_{u2}^i)^2 - \sum_{i=1}^{r}(x_{u2}^i - \log x_{l2}^i)^2 \\ \\ f1 = \sum_{i=1}^{p}(x_{u1}^i)^2 \\ f2 = \sum_{i=1}^{q}(x_{l1}^i)^2 \\ f3 = \sum_{i=1}^{r}(x_{u2}^i - \log x_{l2}^i)^2 \end{array} \tag{15}$$

The range of variables is as follows,

$$\begin{array}{l} x_{u1}^i \in [-5,10], \;\; \forall \;\; i \in \{1,2,\ldots,p\} \\ x_{u2}^i \in [-5,1], \;\; \forall \;\; i \in \{1,2,\ldots,r\} \\ x_{l1}^i \in [-5,10], \;\; \forall \;\; i \in \{1,2,\ldots,q\} \\ x_{l2}^i \in (0,e], \;\; \forall \;\; i \in \{1,2,\ldots,r\} \end{array} \tag{16}$$

Relationship between upper level variables and lower level optimal variables is given as follows,

$$\begin{array}{l} x_{l1}^i = 0, \;\; \forall \;\; i \in \{1, 2, \ldots, q\} \\ x_{l2}^i = \log^{-1} x_{u2}^i, \;\; \forall \;\; i \in \{1, 2, \ldots, r\} \end{array} \tag{17}$$

The values of the variables at the optima are $\mathbf{x}_u = 0$ and $\mathbf{x}_l$ is obtained by the relationship given above. Both the upper and lower level functions are equal to 0 at the optima.

C. SMD3

In this test problem there is a cooperation between the two levels. The difficulty introduced is in terms of multi-modality at the lower level which contains the Rastrigin's function. The upper level is convex with respect to upper level variables and optimal lower level variables.

$$\begin{array}{l} F1 = \sum_{i=1}^{p} (x_{u1}^i)^2 \\ F2 = \sum_{i=1}^{q} (x_{l1}^i)^2 \\ F3 = \sum_{i=1}^{r} (x_{u2}^i)^2 + \sum_{i=1}^{r} ((x_{u2}^i)^2 - \tan x_{l2}^i)^2 \\ \\ f1 = \sum_{i=1}^{p} (x_{u1}^i)^2 \\ f2 = q + \sum_{i=1}^{q} \left(\left(x_{l1}^i\right)^2 - \cos 2\pi x_{l1}^i \right) \\ f3 = \sum_{i=1}^{r} ((x_{u2}^i)^2 - \tan x_{l2}^i)^2 \end{array} \tag{18}$$

The range of variables is as follows,

$$\begin{array}{l} x_{u1}^i \in [-5, 10], \;\; \forall \;\; i \in \{1, 2, \ldots, p\} \\ x_{u2}^i \in [-5, 10], \;\; \forall \;\; i \in \{1, 2, \ldots, r\} \\ x_{l1}^i \in [-5, 10], \;\; \forall \;\; i \in \{1, 2, \ldots, q\} \\ x_{l2}^i \in (\frac{-\pi}{2}, \frac{\pi}{2}), \;\; \forall \;\; i \in \{1, 2, \ldots, r\} \end{array} \tag{19}$$

Relationship between upper level variables and lower level optimal variables is given as follows,

$$\begin{array}{l} x_{l1}^i = 0, \;\; \forall \;\; i \in \{1, 2, \ldots, q\} \\ x_{l2}^i = \tan^{-1} (x_{u2}^i)^2, \;\; \forall \;\; i \in \{1, 2, \ldots, r\} \end{array} \tag{20}$$

The values of the variables at the optima are $\mathbf{x}_u = 0$ and $\mathbf{x}_l$ is obtained by the relationship given above. Both the upper and lower level functions are equal to 0 at the optima.

D. SMD4

In this test problem there is a conflict between the two levels. The difficulty is in terms of multi-modality at the lower level which contains the Rastrigin's function. The upper level is convex with respect to upper level variables and optimal lower level variables.

$$\begin{array}{l} F1 = \sum_{i=1}^{p} (x_{u1}^i)^2 \\ F2 = -\sum_{i=1}^{q} (x_{l1}^i)^2 \\ F3 = \sum_{i=1}^{r} (x_{u2}^i)^2 - \sum_{i=1}^{r} (|x_{u2}^i| - \log(1 + x_{l2}^i))^2 \\ \\ f1 = \sum_{i=1}^{p} (x_{u1}^i)^2 \\ f2 = q + \sum_{i=1}^{q} \left(\left(x_{l1}^i\right)^2 - \cos 2\pi x_{l1}^i \right) \\ f3 = \sum_{i=1}^{r} (|x_{u2}^i| - \log(1 + x_{l2}^i))^2 \end{array} \tag{21}$$

The range of variables is as follows,

$$\begin{array}{l} x_{u1}^i \in [-5, 10], \;\; \forall \;\; i \in \{1, 2, \ldots, p\} \\ x_{u2}^i \in [-1, 1], \;\; \forall \;\; i \in \{1, 2, \ldots, r\} \\ x_{l1}^i \in [-5, 10], \;\; \forall \;\; i \in \{1, 2, \ldots, q\} \\ x_{l2}^i \in [0, e], \;\; \forall \;\; i \in \{1, 2, \ldots, r\} \end{array} \tag{22}$$

Relationship between upper level variables and lower level optimal variables is given as follows,

$$\begin{array}{l} x_{l1}^i = 0, \;\; \forall \;\; i \in \{1, 2, \ldots, q\} \\ x_{l2}^i = \log^{-1} |x_{u2}^i| - 1, \;\; \forall \;\; i \in \{1, 2, \ldots, r\} \end{array} \tag{23}$$

The values of the variables at the optima are $\mathbf{x}_u = 0$ and $\mathbf{x}_l$ is obtained by the relationship given above. Both the upper and lower level functions are equal to 0 at the optima.

E. SMD5

In this test problem there is a conflict between the two levels. The difficulty introduced is in terms of multi-modality and convergence at the lower level. The lower level problem contains the banana function such that the global optimum lies in a long, narrow, flat parabolic valley. The upper level is convex with respect to upper level variables and optimal lower level variables.

$$\begin{array}{l} F1 = \sum_{i=1}^{p} (x_{u1}^i)^2 \\ F2 = -\sum_{i=1}^{q} \left(\left(x_{l1}^{i+1} - \left(x_{l1}^i\right)^2 \right) + \left(x_{l1}^i - 1\right)^2 \right) \\ F3 = \sum_{i=1}^{r} (x_{u2}^i)^2 - \sum_{i=1}^{r} (|x_{u2}^i| - (x_{l2}^i)^2)^2 \\ \\ f1 = \sum_{i=1}^{p} (x_{u1}^i)^2 \\ f2 = \sum_{i=1}^{q} \left(\left(x_{l1}^{i+1} - \left(x_{l1}^i\right)^2 \right) + \left(x_{l1}^i - 1\right)^2 \right) \\ f3 = \sum_{i=1}^{r} (|x_{u2}^i| - (x_{l2}^i)^2)^2 \end{array} \tag{24}$$

The range of variables is as follows,

$$\begin{array}{l} x_{u1}^i \in [-5, 10], \;\; \forall \;\; i \in \{1, 2, \ldots, p\} \\ x_{u2}^i \in [-5, 10], \;\; \forall \;\; i \in \{1, 2, \ldots, r\} \\ x_{l1}^i \in [-5, 10], \;\; \forall \;\; i \in \{1, 2, \ldots, q\} \\ x_{l2}^i \in [-5, 10], \;\; \forall \;\; i \in \{1, 2, \ldots, r\} \end{array} \tag{25}$$

Relationship between upper level variables and lower level optimal variables is given as follows,

$$\begin{array}{l} x_{l1}^i = 0, \;\; \forall \;\; i \in \{1, 2, \ldots, q\} \\ x_{l2}^i = \sqrt{|x_{u2}^i|}, \;\; \forall \;\; i \in \{1, 2, \ldots, r\} \end{array} \tag{26}$$

The values of the variables at the optima are $\mathbf{x}_u = 0$ and $\mathbf{x}_l$ is obtained by the relationship given above. Both the upper and lower level functions are equal to 0 at the optima.

F. SMD6

In this test problem there is a conflict between the two levels. The problem contains infinitely many global solutions at the lower level, for any given upper level vector. Out of the entire global solution set, there is only a single lower level point which corresponds to the best upper level function value.

$$\begin{array}{l} F1 = \sum_{i=1}^{p} (x_{u1}^i)^2 \\ F2 = -\sum_{i=1}^{q} (x_{l1}^i)^2 + \sum_{i=q+1}^{q+s} (x_{l1}^i)^2 \\ F3 = \sum_{i=1}^{r} (x_{u2}^i)^2 - \sum_{i=1}^{r} (x_{u2}^i - x_{l2}^i)^2 \\ \\ f1 = \sum_{i=1}^{p} (x_{u1}^i)^2 \\ f2 = \sum_{i=1}^{q} (x_{l1}^i)^2 + \sum_{i=q+1, i=i+2}^{q+s-1} (x_{l1}^{i+1} - x_{l1}^i)^2 \\ f3 = \sum_{i=1}^{r} (x_{u2}^i - x_{l2}^i)^2 \end{array} \tag{27}$$

The range of variables is as follows,

$$\begin{aligned} x^i_{u1} &\in [-5, 10], \;\; \forall \;\; i \in \{1, 2, \ldots, p\} \\ x^i_{u2} &\in [-5, 10], \;\; \forall \;\; i \in \{1, 2, \ldots, r\} \\ x^i_{l1} &\in [-5, 10], \;\; \forall \;\; i \in \{1, 2, \ldots, q+s\} \\ x^i_{l2} &\in [-5, 10], \;\; \forall \;\; i \in \{1, 2, \ldots, r\} \end{aligned} \quad (28)$$

Relationship between upper level variables and lower level optimal variables is given as follows,

$$\begin{aligned} x^i_{l1} &= 0, \;\; \forall \;\; i \in \{1, 2, \ldots, q\} \\ x^i_{l2} &= x^i_{u2}, \;\; \forall \;\; i \in \{1, 2, \ldots, r\} \end{aligned} \quad (29)$$

The values of the variables at the optima are $\mathbf{x}_u = 0$ and $\mathbf{x}_l$ is obtained by the relationship given above. Both the upper and lower level functions are equal to 0 at the optima.

V. Baseline Solution Methodology

To provide a benchmark for evaluating the performance of alternative bilevel solution methodologies, we solve the above described test problems using a nested scheme. As discussed below, the method relies on a steady state single objective real coded genetic algorithm to solve the problems at both levels. The underlying algorithm at both levels is a modified version of the procedures [9], [8] based on the single objective Parent Centric Crossover (PCX) [4].

A. Bilevel Evolutionary Algorithm

The bilevel evolutionary algorithm designed to handle single objective bilevel problems is a simple minded strategy where the lower level problem is solved for all given upper level points. Whenever the lower level optimal member is to be determined for a new upper level member, the information is utilized from the nearest upper level member for which lower level optimal solution is known. The main steps in the algorithm are summarized as follows:

Step 0: Initialization. The algorithm starts with a random population of size N, which is initialized by generating the required number of upper level variables, and then executing the lower level optimization procedure to determine the corresponding optimal lower level variables. Fitness is assigned based on upper level function value and constraints.

Step 1: Selection of upper level parents. Given the current population, we randomly choose 2μ number of members from the population and perform a tournament selection. This produces μ number of parents.

Step 2: Evolution at the upper level. From the selected parents, the best member is chosen as the index parent. Then, we create λ number of offsprings from the chosen μ parents, using crossover (see description below) and polynomial mutation.

Step 3: Lower level optimization. For an offspring member, the closest upper level member is determined. From the closest upper level member, the lower level optimal member is copied. Thereafter, a lower level optimization run is performed. The lower level is called with a population size n, the copied lower level member from the closest upper level member is included as one of the members in the population. This step is executed for each of the offspring members produced. Fitness is assigned based on upper level function value and constraints.

Step 4: Population Update After optimizing the offsprings, r members are chosen from the parent population. A pool of chosen r members and λ offsprings is formed. The best r members from the pool replace the chosen r members from the population. A termination check is performed and the algorithm moves to the next generation (Step 1) if the termination check (see description below) is false.

B. Lower Level Optimization

The lower level optimization procedure is similar to the upper level procedure. The fitness assignment at this level is performed based on lower level function value and constraints. After initializing n lower level members, 2μ members are randomly chosen from the population. A tournament selection is performed and μ parents are chosen for crossover. The best parent among μ parents is chosen as the index parent and λ number of offsprings are produced using the crossover and mutation operators. A population update is performed as before by choosing r random members from the population. A pool is formed using r chosen members and λ offsprings, from which the best r members are used to replace the r chosen members from the population. Next generation is executed if the termination criteria is not satisfied.

C. Parameters

The parameters in the algorithm are fixed as $\mu = 3$, $\lambda = 2$ and $r = 2$. Crossover probability is fixed at 0.9 and the mutation probability is 0.1. The upper level population size N and the lower level population size n were fixed at 50 for all the test problems. The algorithm uses a constraint handling procedure with a reducing window, however, we do not discuss this here as the problems being handled in this paper are unconstrained problems.

D. Crossover Operator

The crossover operator used in Step 2 is similar to the PCX operator proposed in [8]. The operator creates a new solution from 3 number of parents as follows:

$$\vec{C} = \vec{X}_p + \omega_\xi \vec{D} + \omega_\eta \frac{\vec{P}_2 - \vec{P}_1}{2} \quad (30)$$

The terms used in the above equation are defined as follows:

- $\vec{X}_p$ is the *index* parent
- $\vec{D} = \vec{X}_p - \vec{G}$, where $\vec{G}$ is the mean of μ parents
- $\vec{P}_1$ and $\vec{P}_2$ are the other two parents
- $\omega_\xi = 0.1$ and $\omega_\eta = \sum_{i=1}^{m} \frac{m}{abs(X_p^i - G^i)}$ are the two parameters, where m is the number of dimensions of the problem

The two parameters ω_ξ and ω_η, describe the extent of variations along the respective directions.

TABLE II

FUNCTION EVALUATIONS (FE) FOR THE UPPER LEVEL (UL) AND THE LOWER LEVEL (LL) FROM 11 RUNS.

Pr. No.	Best		Median		Worst	
	Total LL FE	Total UL FE	Total LL FE	Total UL FE	Total LL FE	Total UL FE
SMD1	872403	1256	1724241	2644	2176900	3458
SMD2	1052845	1568	1568099	2404	2217637	3550
SMD3	910597	1324	1483884	2338	1903370	2866
SMD4	586734	780	1187981	1720	1412851	2124
SMD5	1325231	1744	2093391	3010	2688936	3694
SMD6	1325594	1704	2429352	3212	3129367	4050

TABLE III

ACCURACY FOR THE UPPER AND LOWER LEVELS, AND THE LOWER LEVEL CALLS FROM 11 RUNS.

Pr. No.	Median UL Accuracy	Median LL Accuracy	Median LL Calls	$\frac{\text{LL Evals}}{\text{LL Calls}}$
SMD1	0.000034	0.000016	2644	648.52
SMD2	0.000005	0.000005	2404	652.29
SMD3	0.000059	0.000026	2338	652.43
SMD4	0.000026	0.000027	1720	690.26
SMD5	0.000004	0.000003	3010	719.85
SMD6	0.000145	0.000071	3212	768.11

E. Termination Criteria

The algorithm uses a variance based termination criteria. When the value of α, described in the following equation becomes less than α_{stop}, the algorithm terminates.

$$\alpha = \sum_{i=1}^{m} \frac{\sigma^2(x^i_{current})}{\sigma^2(x^i_{initial})}. \tag{31}$$

In the above equation α is restricted between 0 and 1. If the value exceeds 1, it is fixed as 1. m is the dimensionality of the optimization problem being solved, $x^i_{current}$: i $\in \{1, 2, \ldots, m\}$ represents the variables in current population, and $x^i_{initial}$: i $\in \{1, 2, \ldots, m\}$ represents the variables in initial population. The value of α_{stop} is used as $1e-5$ at both the upper and the lower levels. A high accuracy is desired at the lower level because inaccurate lower level solutions may mislead the algorithm in case of a conflict between the two levels.

VI. RESULTS

In this section, we provide the results obtained from solving the proposed test problems using the bilevel evolutionary algorithm. We performed 11 number of runs for each of the test problems with 10 dimensions. For SMD1 to SMD5 we choose $p = 3$, $q = 3$ and $r = 2$, and for SMD6 we choose $p = 3$, $q = 1$, $r = 2$ and $s = 2$.

The results are reported in Table II for best, median, and worst values of function evaluations at upper and lower levels. The accuracy achieved and the number of times lower level optimization was performed in a single execution of the bilevel optimization run are reported in Table III.

VII. CONCLUSIONS

The paper provides a test problem construction procedure for unconstrained single objective bilevel optimization. Using the construction procedure, unconstrained test problems with controllable difficulties can be constructed. A test suit of six bilevel test problems have been proposed which may be used to evaluate the performance of any bilevel algorithm. As a benchmark for comparison, we provide the results from a simple bilevel evolutionary algorithm, which uses a nested strategy to handle bilevel problems. The procedure uses a parent centric crossover based global optimizer at both levels, which successfully solves the test problems. However, the function evaluations required by the algorithm are high, particularly at the lower level, because of the nested nature of the approach.

ACKNOWLEDGMENTS

Authors wish to thank Academy of Finland and Foundation of Helsinki School of Economics for their support of this study.

REFERENCES

[1] P. H. Calamai and L. N. Vicente. Generating linear and linear-quadratic bilevel programming problems. *SIAM J. Sci. Comput.*, 14(1):770–782, 1992.

[2] P. H. Calamai and L. N. Vicente. Generating quadratic bilevel programming test problems. *ACM Trans. Math. Software*, 20(1):103–119, 1994.

[3] B. Colson, P. Marcotte, and G. Savard. Bilevel programming: A survey. *4OR, A Quarterly Journal of Operations Research*, 3(1):87–107, 2005.

[4] K. Deb, A. Anand, and D. Joshi. A computationally efficient evolutionary algorithm for real-parameter optimization. *Evolutionary Computation Journal*, 10(4):371–395, 2002.

[5] S. Dempe. Annotated bibliography on bilevel programming and mathematical programs with equilibrium constraints. *Optimization*, 52(3):339–359, 2003.

[6] A. Mitsos and P. I. Barton. A test set for bilevel programs, 2006. http://yoric.mit.edu/download/Reports/bileveltestset.pdf.

[7] K. Moshirvaziri, M. A. Amouzegar, and S. E. Jacobsen. Test problem construction for linear bilevel programming problems. *Journal of Global Optimization*, 8(3):235–243, 1996.

[8] A. Sinha, A. Srinivasan, and K. Deb. A population-based, parent centric procedure for constrained real-parameter optimization. In *2006 IEEE Congress on Evolutionary Computation (CEC-2006)*, pages 239–245. IEEE Press, 2006.

[9] A. Sinha, S. Tiwari, and K. Deb. A population-based, steady-state procedure for real-parameter optimization. In *2005 IEEE Congress on Evolutionary Computation (CEC-2005)*, pages 514–521. IEEE Press, 2005.